Assisted Excitation of Activations:
A Learning Technique to Improve Object Detectors

Mohammad Mahdi Derakhshani[1*], Saeed Masoudnia[1*], Amir Hossein Shaker[1], Omid Mersa[1],
Mohammad Amin Sadeghi[1], Mohammad Rastegari[2], Babak N. Araabi[1]

[1]MLCM Lab, Department of Electrical and Computer Engineering, University of Tehran, Tehran, Iran.
[2]Allen Institute for Artificial Intelligence (AI2)

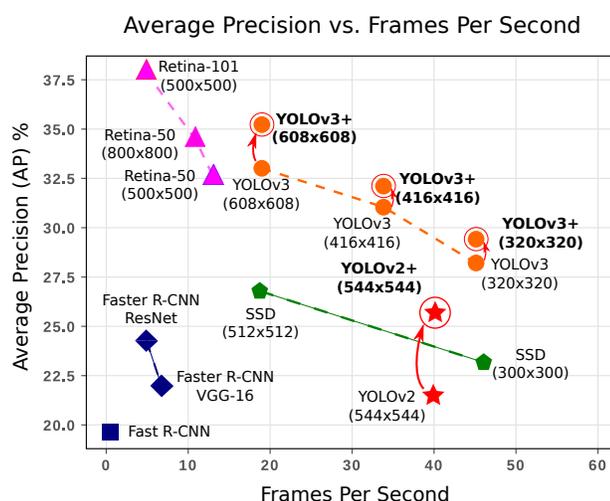Email: mderakhshani, masoudnia, ah.shaaker, o.mersa, asadeghi, araabi{@ut.ac.ir}, mohammadr@allenai.org

Figure 1. Comparison of different object detection algorithms according to their mean Average Precision and speed (Frames Per Second). Our improvements (YOLOv2+ and YOLOv3+, highlighted using circles and bold face type) outperform original YOLOv2 and YOLOv3 in terms of accuracy. In terms of speed, our technique is identical to YOLOv2 and YOLOv3. We have evaluated YOLOv3+ on three different image resolutions.
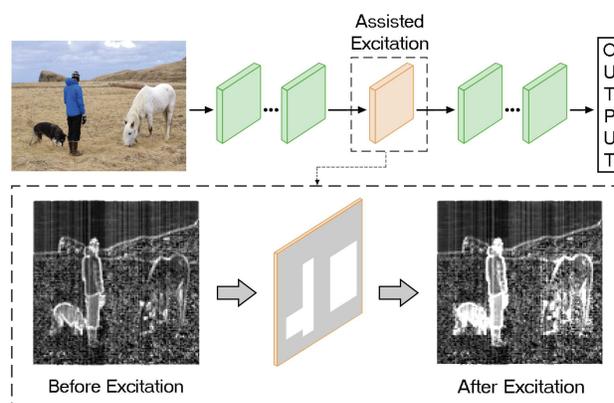

Figure 2. An illustration of our proposed Assisted Excitation Module. We manually excite certain activations during training. These activations help improve localization. We excite activations based on object locations. We applied our technique to YOLO object detectors.

Our technique improves the mAP of YOLOv2 by 3.8% and mAP of YOLOv3 by 2.2% on MSCOCO dataset.This technique is inspired from curriculum learning. It is simple and effective and it is applicable to most single-stage object detectors.

Abstract

We present a simple and effective learning technique that significantly improves mAP of YOLO object detectors without compromising their speed. During network training, we carefully feed in localization information. We excite certain activations in order to help the network learn to better localize (Figure 2). In the later stages of training, we gradually reduce our assisted excitation to zero. We reached a new state-of-the-art in the speed-accuracy trade-off (Figure 1).

1. Introduction

Modern object detectors use Convolutional Neural Networks [22, 29, 30]. Most of modern object detectors fall into one of two categories: Single-stage detectors (YOLO [27, 28, 29], SSD [24] and Retina-Net [22]) and two-stage detectors (R-CNN [13] and variants [12, 30]). Two-stage detectors first generate a number of proposals and then classify them. In contrast, single-stage detectors perform detection in one pass, straight from raw images

_____
*equally contributed

Table 1. Comparison of the architectures and the characteristics of the three versions of YOLO object detector.

Model	Backbone	Structure	Detection Resolution	Detections Per Grid
YOLOv1	Darknet inspired by GoogleNet [33] (without inception module) and NIN [20]	24 convolutional layers followed by 2 fully connected layers	Grid of 7×7	2
YOLOv2	Darknet19 inspired by VGG [31] and NIN [20]	FCN [32] with 19 convolution layers and 5 max-pooling	Grid with stride=32	5
YOLOv3	Darknet53 inspired by ResNet [16] and FPN [21]	FPN with 75 convolutional layers without max-pooling	Grids with strides of 32, 16 and 8	3

to final detections. Figure 1 compares a number of notable object detectors according to speed and accuracy. YOLO (You Only Look Once) [27] is one of the most successful object detector families. These detectors are developed by Redmon et al. [27, 28, 29] in three versions: YOLOv1 (2016) [27], YOLOv2 (2017) [28], and YOLOv3 (2018) [29]. YOLO detectors are fast and accurate at the same time. They work in real-time and produce high-accuracy detections [25].

While YOLO detectors are very successful, they face two challenges: 1- difficulty in localization [27, 28, 29], and 2- foreground-background class imbalance at training [22]. All versions of YOLO face these challenges. In the latest work, Redmon et al. [29] reported: "*The performance drops significantly as the IOU threshold increases, indicating YOLOv3 struggles to get the boxes perfectly aligned with the object.*"

Localization problem occurs because YOLO performs classification and localization simultaneously. The last convolutional layer is typically rich in terms of semantics. This is ideal for classification; however, the last convolutional layer is often spatially course for localization. Thus compared to other successful object detectors, YOLO makes more localization errors.

Unlike two-stage detectors, single-stage detectors do not reduce search space to a limited number of candidate proposals. Instead, their search space includes a large number of possible bounding-boxes (around 10^4 to 10^5). Most of these bounding-boxes are negative examples and most of negative examples are easy to classify. As a result, a detector's loss is overwhelmed with easy negative examples while being trained.

This problem was described by Lin et al. [22] as foreground-background class imbalance problem. They offered "focal loss" to dynamically focus on more difficult negative examples. This loss function greatly improved detection accuracy and resulted in a new model named RetinaNet. Redmon et al. [29] examined focal loss for YOLOv3, however, they reported that focal loss has been unable to improve YOLOv3.

1.1. Overview of our Solution

We propose a solution to address these challenges in YOLO. We only change the way these networks are trained. We propose a technique to *excite* certain activation maps in the network during *training*. We do not change network architecture during inference; we do not change loss function; and we do not manipulate network input or output.

We test our technique on the training of YOLOv2 and YOLOv3 detectors. During the first epochs of training, we manually excite certain activations in feature maps. Then, in the later epochs of training, we gradually reduce excitation levels to zero. During the last epochs of training, we stop exciting activations. Therefore, the network learns to perform detection without assisted excitation. This strategy is inspired by curriculum learning [2]; it simplifies the task of detection and localization in the early stages of training and gradually makes the task more difficult and realistic.

We excite activations corresponding to object locations (extracted from ground truth) in feature maps. While we excite these activations, detection becomes easier because our model receives feedback from ground-truth. Therefore, we argue that these excitations help the network 1- improve localization and 2- focus on hard negatives rather than easy negatives. We refer to our method as Assisted Excitation (AE) because we manually excite activations to assist with training.

Our technique helps YOLOv2 improve by 3.8% mAP and YOLOv3 by 2.2% mAP on MSCOCO, without any loss of speed.

2. Related Works

YOLO: Through a sequence of advances, Redmon et al. [27, 28, 29] proposed three versions of YOLO. The performance of the latest model is on par with the state-of-the-art. Moreover, YOLO sits at the faster end of the speed-accuracy trade-off. We briefly compare the architecture and characteristics of these versions of YOLO in Table 1.

Augmenting auxiliary information into CNNs: Introducing auxiliary information into CNNs has shown to be useful in certain applications [35, 26, 3, 19, 17]. A number of works concluded that joint learning of object detection and semantic segmentation can improve both results. These works fall into two categories. The first
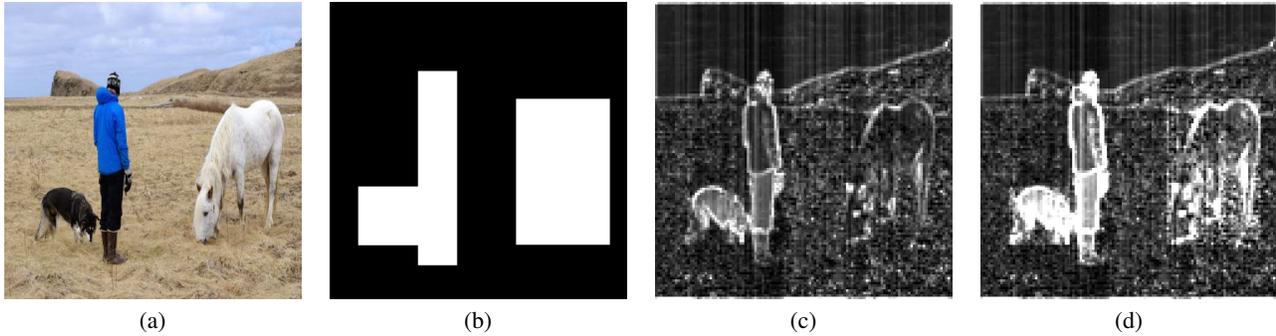
Figure 3. Illustration of our Assisted Excitation process. (a) Reference image; (b) Map of object bounding-boxes used to mask excitations; (c) Averaged activation before Assisted Excitation Layer; (d) Averaged activation after Assisted Excitation Layer. Please note that excited locations correspond to the object map.

category [14, 6] attempts to perform simultaneous detection and segmentation and improve the performance of both tasks [14, 6, 8, 34, 5]. This combined task is known as instance-aware semantic segmentation.

The second category [14, 15, 10, 11, 36] aims to only boost object detection by introducing segmentation features. Gidaris and Komodakis [11] added semantic segmentation-aware CNN features to detection features at the highest level of R-CNN model. Their model used the auxiliary segmentation information to refine localization. He et al. [15] proposed Mask R-CNN which extends Faster R-CNN [30].They added a branch for predicting an object segmentation mask in parallel with the existing detection branch. Zhang et al. [36] extended an SSD-based object detection model by adding a segmentation branch. However, this branch was trained by weak segmentation ground-truth (box-level segmentation), thus no extra annotation was required.

Several works [7, 4] used the approach of joint segmentation and detection in the application of pedestrian detection. Brazil et al. [4] also offered multi-task learning on pedestrian detection and semantic segmentation based on the extension of R-CNN. In this model, the weak box-based segmentation mask is infused with both stages of R-CNN model.

Among the reviewed studies, our proposed method is more related to [4, 36]. Similar to their approaches, we also employ weak segmentation ground-truth only during training and the model efficiency is not affected in our inference phase. Another similarity lies in the fact that there is no need for extra annotation rather than weakly annotated boxes in the detection annotation.

Although the previous studies [4, 36] developed their models based on R-CNN and SSD respectively, our model is built on top of YOLO model. These studies augmented auxiliary segmentation layers with an extra loss function. Our proposed method does not impose extra computational burden in the training phase. Our main novelty lies in the way of incorporating the ground truth information into the CNN.

3. Challenges in Single-stage Detectors

In Section 1.1 we described two challenges that YOLO architecture faces. Here we describe them in more details:

1. Localization Problem: For the sake of speed, YOLO performs localization and classification at the same time. Final layers of YOLO architecture produce high-level feature maps. These feature maps are ideal for classification because they are semantic and high-level. However, They are not ideal for localization because they are spatially too course. YOLOv3 tries to address this problem by passing on low-level features (from earlier stages) into localization process. However, Redmon et al. acknowledge that all three versions of YOLO suffer from localization problem.

2. Foreground-Background class imbalance problem: Two-stage detectors first identify a limited number of object proposals and then classify them. The first stage takes care of most of the localization task. Therefore, the search space in the second stage is limited to a number of proposals that have proper localization.

   In contrast, single-stage detectors need to search through a large number possible bounding-boxes (10^4 to 10^5). Many of these bounding-boxes include an object, but most of those containing an object are not localized properly. Therefore, the detector has to search through all of these bounding-boxes and find the single bounding-box that localizes the object the best. this problem is described by Lin et. al. [22] They propose a new loss function to address this problem. Redmon et al. [29]examined focal loss for YOLOv3, however, it did not work out.
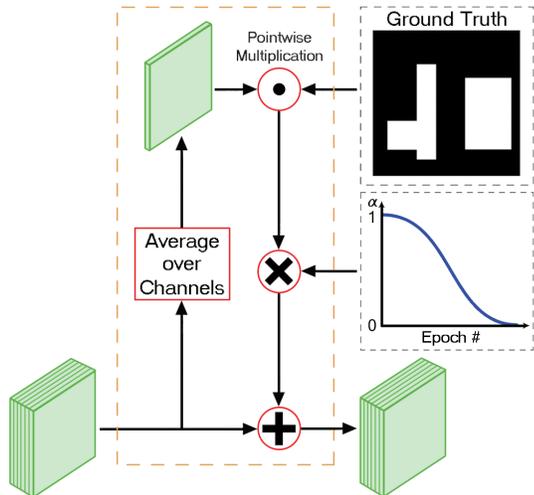
Figure 4. Assisted excitation layer: This layer takes an activation tensor as input. It first averages out all activation maps in input tensor. Then, it masks the results according to object bounding box locations. The excitation value is multiplied by Excitation factor α. The result is finally added to each channel of the input tensor and is passed on to the next layer.

4. Assisted Excitation Process

We propose a technique to address these challenges. Our technique only applies to the *learning process*. We neither change network architecture nor we change the detection process.

During training, we manually excite certain activations corresponding to object locations. During the initial epochs of training, we perform this additional excitation, however, we gradually decrease the excitation level in the later epochs to zero, see Figure 4.

In the initial epochs of training, our manual activation gives a boost to the best localization bounding-box. This activation helps distinguish the best bounding-box from slightly misplaced bounding-boxes. As we decrease excitation level during the next epochs of training, our model continues to distinguish the best bounding-box from misplaced ones.

We manually excite activations at the locations that we know some object exists. We know where objects exist from the ground-truth annotation, see Figure 3. Ground-truth information is known only during training. Therefore, our final trained model cannot depend on ground-truth. Since we stop manual excitation in the latest stages of training, our model learns to work independent of ground-truth. However, during the initial stages of training, our model depends on a manual excitation that is guided by ground-truth.

These excitations guide the model to 1- improve localization and 2- focus on hard negatives rather than easy negatives. We call our proposed method as Assisted Excitation.

Our technique falls into curriculum learning framework described by Bengio et al. [2]. The idea behind curriculum learning is that learning space is non-convex, and learning can fall into a bad local-minima. They argue that if we first learn easier tasks and the continue with more complex tasks, we get better performance in terms of the quality of local-minima and generalization.

4.1. Assisted Excitation using Ground-Truth

Assisted excitation can be viewed as a network layer that manipulates neural activations. We can describe an assisted excitation module as follows:

$$a^{l+1}_{(c,i,j)} = a^{l}_{(c,i,j)} + \alpha(t)\, e_{(c,i,j)} \tag{1}$$

where a^l and a^{l+1} are activation tensors at levels l and $l+1$. e is excitation tensor and α is excitation factor that depends on epoch number t. Also (c, i, j) refer to channel number, row and column. During training, $\alpha(t)$ starts with a non-zero value for initial epochs and gradually decays to zero. e is a function of a_l and ground-truth. To compute e, we first construct a bounding-box map g as follows:

$$g_{(i,j)} = \begin{cases} 1, & \text{If some bbox exists at cell}(i,j) \\ 0, & \text{If no bbox exists at cell}(i,j). \end{cases}$$

The excitation e in bbox locations can be applied based on different strategies. The straight forward excitation strategy is as follows:

$$e_{(c,i,j)} = \frac{g_{(i,j)}}{d} a(c,i,j) \tag{2}$$

This strategy excites the activation of bbox location in each channel. Alternative strategy can inhibit out of bbox locations which makes the activations in the bbox locations relatively highlighted.

$$e_{(c,i,j)} = -(1 - g_{(i,j)})a_{(c,i,j)} \tag{3}$$

These two strategies highlight the activation of bbox locations in each channel independently. We have tried a few variants of this excitation strategy. However, the best performance is not achieved based on these independent manipulation but with the excitation by shared information of bbox locations over all channels. In our method, $e_{(c,i,j)}$ takes an average over all channels of $a^{l}_{(c,i,j)}$. Therefore, it is identical for all values of c. We compute excitation tensor e as follows:

$$e_{(c,i,j)} = \frac{g_{(i,j)}}{d} \sum_{c=1}^{d} a_{(c,i,j)} \tag{4}$$

where d refers to the number of feature channels. All the mentioned strategies improve localization. However, the last strategy (Eq 4) outperformed the others.

$$\alpha(t) = .5 \times \frac{1 + Cos(\pi.t)}{Max\_Iteration} \tag{5}$$

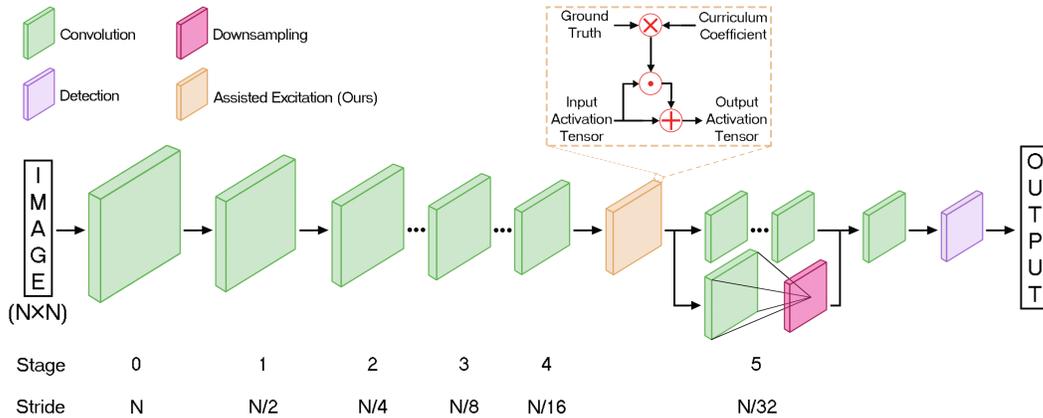Figure 4 illustrates our AE layer in more details.

Figure 5. YOLOv2+ architecture. YOLOv2 architecture is modified with our new assisted excitation layer. AE can be added at the end of each stage; Our experiments show that the end of stage 4 is the optimal place for AE. Each stage is composed of a series of activation tensors which have similar resolutions. For example, assume that the input image size is 480x480. Stage 1, stage 2, stage 3, stage 4, stage 5 and stage 6 contain tensors with resolutions 240x240, 120x120, 60x60, 30x30 and 15x15 respectively.
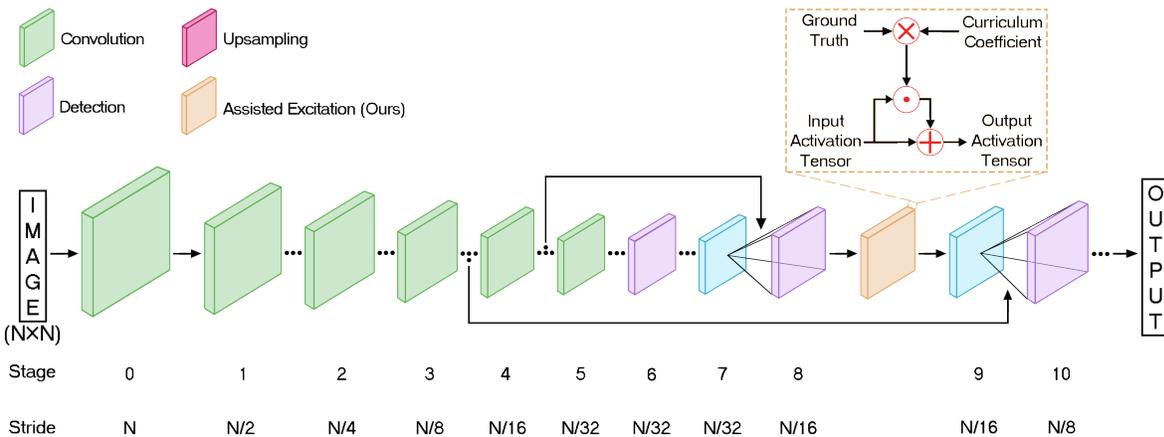

Figure 6. YOLOv3+ architecture. YOLOv3 architecture, which was inspired by [1], is augmented with an assisted excitation layer. The new layer is added to the end of stage 8.

4.2. Inference

During inference, $\alpha = 0$ and the output of AE layer is identical to its input. Therefore, AE layer is essentially removed during inference. During the final epochs of training, our model learns to function without requiring input from ground-truth. Therefore, we do not use ground-truth information.

In practice, our model architecture is identical to YOLO during inference. Our trained model differs from the standard YOLO model only in model weights. This has two major benefits:

1. Our trained model is plug and play. We can reuse the heavily optimized detectors developed for all devices.

2. Our inference time remains identical to the original YOLO detectors while we get better accuracy.

4.3. Assisted Excitation in YOLOv2 and YOLOv3

We used Assisted Excitation in YOLOv2 and YOLOv3. For each of the detectors, we performed an ablation study to examine the improvement if we place AE at each stage. We report the results in Experiments section. Figure 5 illustrates the optimal stage for AE in YOLOv2 architecture. Figure 6 illustrates the optimal stage for AE in YOLOv3 architecture.

5. Experiments and Results

Datasets: We applied our technique on YOLOv2 and YOLOv3. We evaluated the techniques using two benchmarks: MSCOCO [23] and PASCAL VOC 2007, 2012 [9]. Similar to the convention of the original YOLO papers [28, 29], we compare YOLOv2+ with YOLOv2, on PASCAL
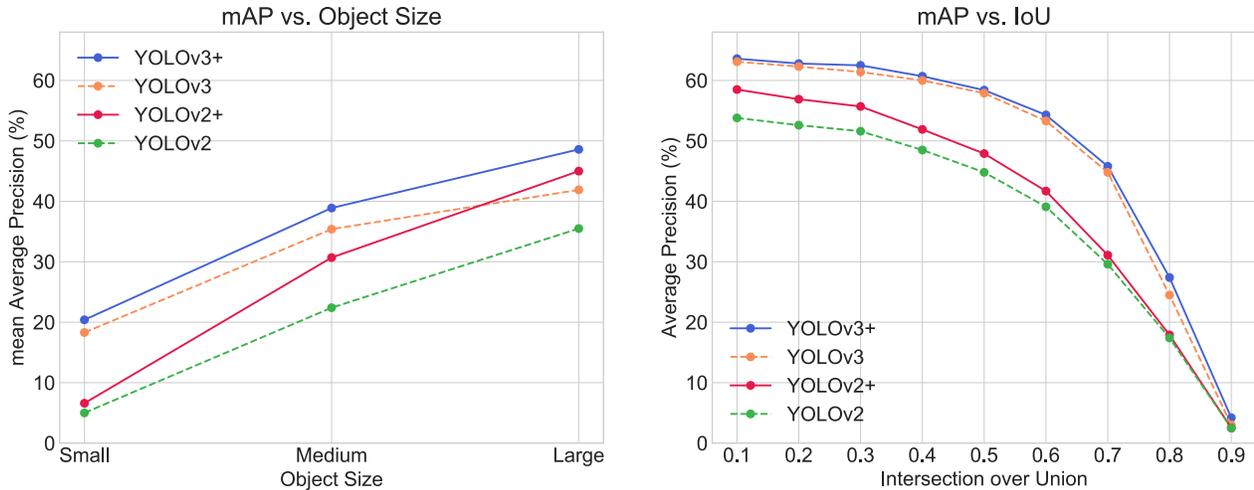
Figure 7. Left: comparison of our proposed methods, YOLOv2+ and YOLOv3+, with their baselines, YOLOv2 and YOLOv3, based on prediction size. As shown, the larger an object is, the more improvement we obtain. Right: comparison of our proposed methods, YOLOv2+ and YOLOv3+, with their baselines, YOLOv2 and YOLOv3, based on Intersection over Union (IoU) threshold.

Table 2. The results of applying AE module through different stages of YOLOv2+. Our proposed model significantly improved the accuracies where applied on the different stages. However, the best accuracy in all terms of AP is achieved in the stage 4.

Method	Stage	AP	AP_{50}	AP_{75}
YOLOv2	-	21.6	44.0	19.2
YOLOv2+ (480)	stage 2	24.6	44.8	24.6
YOLOv2+ (480)	stage 3	25	46	24.9
YOLOv2+ (480)	stage 4	**25.4**	**46.9**	**25.1**

Table 3. The results of different AE strategies in YOLOv2+.

Method	Strategy	AP	AP_{50}	AP_{75}
YOLOv2		21.6	44.0	19.2
YOLOv2+ (544)	strategy in Eq. 2	25.1	45.8	25.8
YOLOv2+ (544)	strategy in Eq. 3	24.8	45	25
YOLOv2+ (544)	strategy in Eq. 4	**26**	**47.9**	**25.8**

VOC 2007, 2012 and MSCOCO 2014. Also, we compare YOLOv3+ with YOLOv3 on MSCOCO 2017. Moreover, we also compare with other state-of-the-art detectors on these datasets.

Training: For training, we trained YOLOv2+ and YOLOv3+ from scratch according to the best practices in their original studies [28, 29]. We used Darknet19 [28] and Darknet53 [29] that were pre-trained on IMAGENET dataset, as backbones. Then, we trained whole architectures using Adam [18] with initial learning rate of 10^{-5}, weight decay of 0.0005, and batch size of 48.

Table 4. The comparison results of YOLOv2+ with YOLOv2 and the other state-of-the-art detectors on MSCOCO test dev-set 2015. The results for the other methods were adapted from. Our proposed YOLOv2+ achieved better accuracies in all terms of APs compared to the previous state-of-the-art detection results.

Method	data	AP	AP_{50}	AP_{75}
Fast RCNN [12]	train	19.7	35.9	-
Faster RCNN [30]	trainval	24.2	45.3	23.5
SSD512 [24]	trainval35k	26.8	46.5	27.8
YOLOv2 [28] (544)	trainval35k	21.6	44.0	19.2
YOLOv2+ (480)	trainval35k	25.4	46.9	25.1
YOLOv2+ (544)	trainval35k	26	47.9	25.8
YOLOv2+ (608)	trainval35k	**27**	**50.9**	**26**

5.1. YOLOv2+

In order to figure out which layer is the optimal place for our Assisted Excitation module, we performed an ablation study. Table 2 lists the accuracy of YOLOv2+ with Assisted Excitation module placed in different stages.

The best accuracy in all terms of AP is achieved when Advanced Excitation is place in stage 4. We also examined different excitation strategies discussed in Section 4.1. As shown in Table 3, the AE strategy in Eq. 4 achieved the best result. We will further discuss the results. In the following experiments, we use this configuration (AE on stage 4) as the default configuration of YOLOv2+.

Based on this setting, we compare YOLOv2+ with YOLOv2 and other current state-of-the-art detectors on MSCOCO test dev-set 2015. The results are compared in Table 4.

We compare YOLOv2+ with YOLOv2 using different

Table 5. The results for comparison of YOLOv2+ with YOLOv2 in different input resolutions on PASCAL VOC 2007 and 2012. These results were also compared with state-of-the-art detectors on this dataset. Our proposed model significantly improved the accuracy of YOLOv2 in all tested resolutions. YOLOv2+ also achieved high accuracy compared to the previous state-of-the-art detection results.

Detection Frameworks		Train	mAP	FPS
Fast R-CNN		2007+2012	70.0	44.0
Faster R-CNN ResNet		2007+2012	76.4	48.4
YOLO		2007+2012	63.4	26.7
SSD500		2007+2012	76.8	26.7
YOLOv2	(416)	2007+2012	76.8	26.7
YOLOv2	(480)	2007+2012	77.8	26.7
YOLOv2	(544)	2007+2012	78.6	26.7
YOLOv2+	(416)	2007+2012	80.6	26.7
YOLOv2+	(480)	2007+2012	81.7	26.7
YOLOv2+	(544)	2007+2012	**82.6**	**26.7**

image resolutions on PASCAL VOC 2007 and VOC 2012. Table 4 compares our results with state-of-the-art works on PASCAL. Table 5 lists more comprehensive detection results for different resolutions in PASCAL VOC 2007 and 2012.

5.2. YOLOv3+

Similar to the original YOLOv3 paper [29], we conducted several experiments on MSCOCO2017 test-dev dataset. We first report our ablation study on placing Assisted Excitation module on different stages of YOLOv3+.

We compared YOLOv3+ with YOLOv3 in Table 6. As shown in the results, the best performance was achieved where Assisted Excitation module is placed on stage 4. In the remaining experiments, we place AE module on stage 4. Based on this setting, we also compare different image resolutions. Table 7 compares YOLOv3+ and YOLOv3 on different input resolutions.Table 8 compares our proposed YOLOv3+ with state-of-the-art detectors on MSCOCO2017 test dev-set.

5.3. Localization

Improvement in localization can be seen in qualitative results. Figure compares localization results between YOLOv2 and YOLOv2+. Figure compares localization results between YOLO32 and YOLOv3+.

Figure 7 right, compares YOLOv2, YOLOv3, YOLOv2+ and YOLOv3+ in terms of mAP versus intersection over union threshold. Figure 7 left, show that our improvement rates increase as the objects become larger. These results in addition to the mentioned theoretical analysis implies that the proposed technique improves the localization ability of YOLO, in specific on medium and large objects.
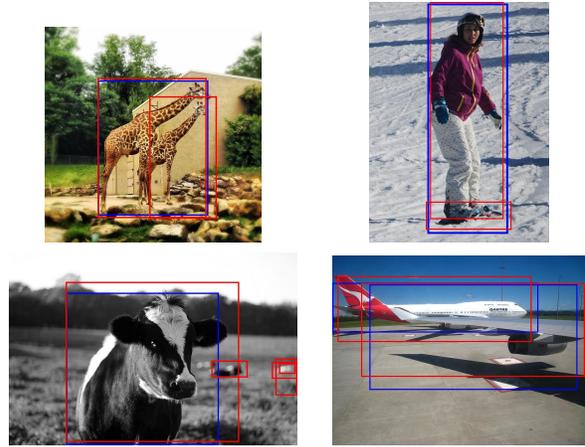

Figure 8. Visual Comparison of YOLOv2+ and YOLOv2 prediction. As shown, Our proposed method (red bounding box) localizes objects better with respect to YOLOv2's prediction (blue bounding box). In addition to localization, our proposed method increases the number of true positive bounding boxes.
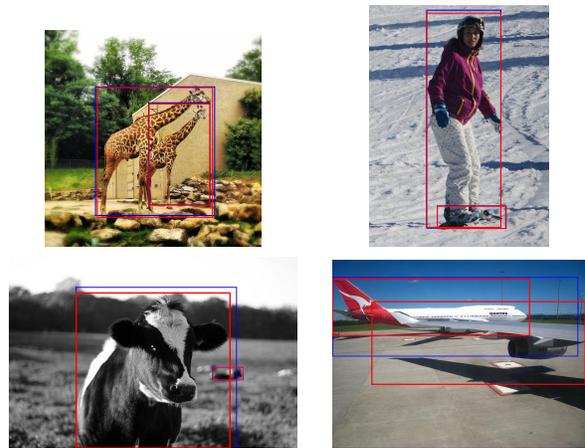

Figure 9. Visual Comparison of YOLOv3+ and YOLOv3 prediction. As shown, Our proposed method (red bounding box) localizes objects better with respect to YOLOv3's prediction (blue bounding box).

Our experimental results show that the AE technique improved the accuracy regardless of what stage it is placed at. Further, our experiments show that best improvements are achieved were AE is placed in the mid-level stages.

In YOLOv2+, the best performance was achieved by placing AE in stage 4 (stride=16). This stage is located in the mid-layers of the model including both localization information and semantic information. In YOLOv3+, the best performance was achieved by placing AE in stage 3 (stride=8). This stage is also located in the mid-layers of the model. The excitation in this stage affects not only in the first detection head but also in both second and third heads because of the skip connections.

Table 6. The comparison results of YOLOv2+ with state-of-the-art detectors on PASCAL VOC 2012. The results for the other detectors were adapted from [28]. Our proposed YOLOv2+ achieved better accuracies in all terms of APs compared to the previous state-of-the-art detection results.

Method	mAP	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
Fast R-CNN	68.4	82.3	78.4	70.8	52.3	38.7	77.8	71.6	89.3	44.2	73.0	55.0	87.5	80.5	80.8	72.0	35.1	68.3	65.7	80.4	64.2
Faster R-CNN	70.4	84.9	79.8	74.3	53.9	49.8	77.5	75.9	88.5	45.6	77.1	55.3	86.9	81.7	80.9	79.6	40.1	72.6	60.9	81.2	61.5
YOLO	57.9	77.0	67.2	57.7	38.3	22.7	68.3	55.9	81.4	36.2	60.8	48.5	77.2	72.3	71.3	63.5	28.9	52.2	54.8	73.9	50.8
SSD512	74.9	87.4	82.3	75.8	59.0	52.6	81.7	81.5	90.0	55.4	79.0	59.8	88.4	84.3	84.7	83.3	50.2	78.0	66.3	86.3	72.0
YOLOv2 544	73.4	86.3	82.0	74.8	59.2	51.8	79.8	76.5	90.6	52.1	78.2	58.5	89.3	82.5	83.4	81.3	49.1	77.2	62.4	83.8	68.7
YOLOv2+ 544	**75.6**	**87.9**	**85.1**	**76.1**	**62.0**	**53.7**	**81.2**	**79.2**	**93.1**	**53.9**	**81.1**	**59.4**	**90.6**	**84.7**	**85.6**	**84.7**	**51.4**	**79.8**	**64.7**	**86.7**	**71.3**

Table 7. The results for applying AE module in different stages of YOLOv3+ on MSCOCO2017 test dev-set. These results were compared with original YOLOv3. Our proposed YOLOv3+ improved the accuracies in all tested stages. However, the best accuracies were achieved for stage 3.

Excitation Stage	Stage	AP	AP_{50}	AP_{75}
YOLOv3 (608)	-	33.0	57.9	34.4
YOLOv3+ (608)	Stage3	**35.2**	**58.4**	**38.4**
YOLOv3+ (608)	Stage4	35.1	58.2	38.4
YOLOv3+ (608)	Stage5	34.2	56.1	37.6
YOLOv3+ (608)	Stage7	34.5	58.0	37.9
YOLOv3+ (608)	Stage9	33.5	54.6	37.1

Table 8. Ablation study on improvement of YOLOv3+ in different input resolutions.

Method	AP	AP_{50}	AP_{75}
YOLOv3 (320)	28.2	47.7	30.0
YOLOv3+(320)	29.1	50.2	30.8
YOLOv3 (416)	31.0	51.0	34.1
YOLOv3+(416)	32.0	53.0	34.8
YOLOv3 (480)	31.6	51.2	34.5
YOLOv3+(480)	32.4	53.0	35.2
YOLOv3 (544)	33.1	51.8	35.9
YOLOv3+(544)	**33.8**	**55.5**	**37.3**

Table 9. The comparison results of YOLOv3+ with state-of-the-art detectors on MSCOCO2017 test dev-set. The results for the other detectors were adapted from [29, 22].Our proposed YOLOv2+ achieved better accuracies in all terms of APs compared to the previous state-of-the-art detection results.

Method	data	AP	AP_{50}	AP_{75}
Faster RCNN+++ [16]	train	34.9	55.7	37.4
Faster RCNN w FPN [21]	train	36.2	59.1	39.0
RetinaNet (800) [22]	trainval35k	**40.8**	**61.1**	**44.1**
YOLOv3 (608) [29]	trainval35k	33.0	57.9	34.4
YOLOv3+ (608)	trainval35k	35.2	58.4	38.4

6. Discussions

Excite object regions vs suppress non-object regions?
We discussed foreground-background class imbalance problem in Section 3. According to this problem, bulk of our search space consists of negative examples. We proposed different object excitations vs non-object suppression strategies in Section 4.1. If we suppress non-object regions, we will affect a large fraction of search space. After we reduce curriculum factor to zero at the end of training, the network will need to re-score most of the candidates in search space. In contrast, when we only excite object regions, the network will only need to keep track of much fewer positive examples. Therefore, the model can more easily handle such a change and yield better results, as shown and compared in Table 3.

What happens during back-propagation?
Our Assisted Excitation module has an effect on back-propagation. Since AE amplifies certain activations, the effect of the receptive field gets amplified as well. Therefore, Positive examples and mislocalized examples will have a higher effect on training (in contrast to easy negative examples that will have lower effect). This is similar to the idea behind Focal Loss. The authors show that increasing focus on positive and hard negative examples improves accuracy.

Curriculum learning
Our technique is similar to curriculum learning because we start from an easier task and gradually move toward more complex tasks. However, there is a subtle difference here. Curriculum learning moves from easy to difficult by introducing increasingly difficult examples. In contrast, we move from easy to difficult by first injecting ground-truth information to the model and gradually removing this information. In other words, our tasks are easier in the initial stages not because the examples are easier, but because we help boost the correct answer. This version of curriculum learning has room to be investigated in further applications.

Applicability
Our technique is applicable not only to other single-stage detectors, but also to two-stage detectors. Moreover, the AE module can be integrated in different CNN architectures for different computer vision problems, e.g., image classification(Fine-grained), segmentation, and synthesis.

References

[1] What's new in yolo v3? towards data science, Apr 2018.

[2] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48. ACM, 2009.

[3] Simone Bianco. Large age-gap face verification by feature injection in deep networks. *Pattern Recognition Letters*, 90:36–42, 2017.

[4] Garrick Brazil, Xi Yin, and Xiaoming Liu. Illuminating pedestrians via simultaneous detection & segmentation. *arXiv preprint arXiv:1706.08564*, 2017.

[5] Jiale Cao, Yanwei Pang, and Xuelong Li. Triply supervised decoder networks for joint detection and segmentation. *arXiv preprint arXiv:1809.09299*, 2018.

[6] Jifeng Dai, Kaiming He, and Jian Sun. Instance-aware semantic segmentation via multi-task network cascades. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3150–3158, 2016.

[7] Xianzhi Du, Mostafa El-Khamy, Jungwon Lee, and Larry Davis. Fused dnn: A deep neural network fusion approach to fast and robust pedestrian detection. In *Applications of Computer Vision (WACV), 2017 IEEE Winter Conference on*, pages 953–961. IEEE, 2017.

[8] Nikita Dvornik, Konstantin Shmelkov, Julien Mairal, and Cordelia Schmid. Blitznet: A real-time deep network for scene understanding. In *ICCV 2017-International Conference on Computer Vision*, page 11, 2017.

[9] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

[10] Sanja Fidler, Roozbeh Mottaghi, Alan Yuille, and Raquel Urtasun. Bottom-up segmentation for top-down detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3294–3301, 2013.

[11] Spyros Gidaris and Nikos Komodakis. Object detection via a multi-region and semantic segmentation-aware cnn model. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1134–1142, 2015.

[12] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[13] Ross B. Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*, pages 580–587, 2014.

[14] Bharath Hariharan, Pablo Arbeláez, Ross Girshick, and Jitendra Malik. Simultaneous detection and segmentation. In *European Conference on Computer Vision*, pages 297–312. Springer, 2014.

[15] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, pages 2980–2988. IEEE, 2017.

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[17] Pan He, Weilin Huang, Tong He, Qile Zhu, Yu Qiao, and Xiaolin Li. Single shot text detector with regional attention. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3047–3055, 2017.

[18] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

[19] Hongyang Li, Jiang Chen, Huchuan Lu, and Zhizhen Chi. Cnn for saliency detection with low-level feature integration. *Neurocomputing*, 226:212–220, 2017.

[20] Min Lin, Qiang Chen, and Shuicheng Yan. Network in network. *arXiv preprint arXiv:1312.4400*, 2013.

[21] Tsung-Yi Lin, Piotr Dollár, Ross B. Girshick, Kaiming He, Bharath Hariharan, and Serge J. Belongie. Feature pyramid networks for object detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 936–944, 2017.

[22] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 2999–3007, 2017.

[23] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[24] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[25] Ajeet Ram Pathak, Manjusha Pandey, and Siddharth Rautaray. Deep learning approaches for detecting objects from images: A review. In Prasant Kumar Pattnaik, Siddharth Swarup Rautaray, Himansu Das, and Janmenjoy Nayak, editors, *Progress in Computing, Analytics and Networking*, pages 491–499, Singapore, 2018. Springer Singapore.

[26] Nazneen Fatema Rajani and Raymond J Mooney. Stacking with auxiliary features. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 2634–2640. AAAI Press, 2017.

[27] Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 779–788, 2016.

[28] Joseph Redmon and Ali Farhadi. YOLO9000: better, faster, stronger. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 6517–6525, 2017.

[29] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *CoRR*, abs/1804.02767, 2018.

[30] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 91–99, 2015.

[31] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[32] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014.

[33] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[34] Marvin Teichmann, Michael Weber, Marius Zoellner, Roberto Cipolla, and Raquel Urtasun. Multinet: Real-time joint semantic reasoning for autonomous driving. In *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 1013–1020. IEEE, 2018.

[35] Tao Wang. Context-driven object detection and segmentation with auxiliary information. 2016.

[36] Zhishuai Zhang, Siyuan Qiao, Cihang Xie, Wei Shen, Bo Wang, and Alan L Yuille. Single-shot object detection with enriched semantics. Technical report, Center for Brains, Minds and Machines (CBMM), 2018.